i.MX233 Linux BSP

Dev Tools specific to iMX233

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Agenda

- Boot media overview
- Prepare to boot from SD/MMC card
- Prepare to boot from Nand flash with rootfs on Nand flash
- Preparation to boot from Nand flash with rootfs on NFS
- Flashing the images using the UUT
- Boot preps (Clock, Power, SDRAM, and Linux)

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Bootstrapping Overview

The Linux kernel image has clock prep, power prep, SDRAM prep and Linux prep added to create a ROM readable image, stmp378x_linux.sb.

- run **‘./ltib –p boot_stream –f**’ to make the boot stream images,
  - stmp378x_linux.sb = elftosb2(clock prep + power prep + sdram prep + linux prep + zImage)
  - stmp378x_uboot.sb = elftosb2(clock prep + power prep + sdram prep + u-boot)
  - uImage = makeimg(zImage)

Typical boot modes for iMx233

- Firmware recovery mode
  - Boot mode 0, set boot switches = off,off,off,off
  - Executes ROM code to talk to Host PC in USB HID class
  - Not used for booting Linux kernel
  - Special applications work with PC

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Bootstrapping Overview (cont.)

Typical boot modes for iMx233 (cont.)

- Nand boot mode
  - Boot mode 4, set boot switches = off,on,off,off
  - ROM reads and executes a boot stream image from Nand Flash
    - ROM reads and executes ‘stmp378x_linux.sb’ from Nand flash
    - ROM uses the rootfs from Nand flash

  - To flash the ‘stmp378x_linux.sb’ and rootfs, the EVK must be booted from either the NFS, SD/MMC card, or run the USB recovery mode with EVK connected to a PC Host running the Universal Updater Tool (UUT).

  - ROM reads and executes ‘stmp378x_uboot.sb’ from Nand flash and boots up the u-boot.
    - U-boot communicate with host over Ethernet or USB using the Trivia File Transfer Protocol to load the ‘uImage’ from the host.
    - U-boot executes the uImage to boot up the Linux kernel and then get the rootfs from the NFS server over the Ethernet or USB.
    - The host can update the files on rootfs which can be seen immediately on the target.
  - To flash the ‘stmp378x_uboot.sb’, the EVK must be booted from either the SD/MMC card, or run the USB recovery mode with EVK connected to a PC Host running the Universal Updater Tool (UUT).

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Bootstrapping Overview (cont.)

Typical boot modes for iMx233 (cont.)

- SD/MMC boot mode
  - Boot mode 9, set boot switches to on,off,off,on
  - The SD/MMC card must have two partitions
    - boot partition
      - partition 1 is 16MB
      - partiton type 53h
      - contains MBR + stmp378x_linux.sb
    - rootfs and data partition
      - partition 2 tooks the remainder disk space
      - Linux ext2 partition type
  - Must have the OTP bit blown for the ROM to read the MBR from the card. This OTP bit is already blown on the EVK.
  - The ROM checks the OTP bit then reads and executes the 'stmp378x_linux.sb' to boot up the device.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Preparing for the SD/MMC boot

Creating two partitions on the SD/MMC card:-

- insert the card and run ‘mount’ command to find out how the card is mounted, e.g. /dev/sdb1
- un-mount the device
- run ‘sudo fdisk /dev/sdb’ if it was earlier mounted as ‘/dev/sdb1’
  - remove the old partitions
    - repeat the ‘d’ command with partition number
  - create a new primary partition 1,
    - press ‘n’, press ‘p’, and then ‘1’
    - press Enter to accept starting block number 1
    - enter ‘16MB’ for the partition size
  - change the partition to type 53h
    - press ‘t’ then enter 53 and Enter
  - Create the second partition
    - press ‘n’, ‘p’, and then ‘2’
    - accept the starting block and size
    - the partition type is default to ext2 type
  - Write the new partition to card and exit
    - press ‘w’

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Preparing for the SD/MMC boot (cont.)

Creating mmc boot image

- Create a file with null in the first 4x512 bytes
  - run ‘dd if=/dev/zero of=mmc_boot_partition.raw bs=512 count = 4’.
- Attach the ‘stmp378x_linux.sb’ boot stream to the file
  - run ‘dd if=~/ltib/rootfs/boot/stmp378x_linux.sb of=mmc_boot_partition.raw ibs=512 seek=4 conv=sync,notrunc’
- Copy the mmc image file to partition 1 of the card
  - run ‘sudo dd if=mmc_boot_partition.raw of=/dev/sdb1’

- Format partition 2 of the card
  - run ‘sudo mkfs.ext2 /dev/sdb2’
- mount the partition 2 as mmc device
  - run ‘sudo mkdir /mnt/mmc’ if there is none
  - run ‘sudo mount /dev/sdb2 /mnt/mmc’
- Copy the rootfs to partition 2
  - run ‘sudo cp –a ~/ltib/rootfs/* /mnt/mmc’

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Preparing for the SD/MMC boot (cont.)

Creating Nand boot images (optional)

- Copy the ‘stmp378x_linux.sb’ to root directory
    - run ‘sudo cp ~/ltib/rootfs/boot/stmp378x_linux.sb /mnt/mmc/root/’
- The rootfs for the Nand flash should not have the images, remove them from the root, i.e. the boot directory of the Nand’s rootfs is empty.
    - run ‘cd ~/ltib’
    - run ‘sudo cp –a rootfs/* myrootfs/*’
    - run ‘sudo rm myrootfs/boot/*’
    - run ‘sudo tar jcf /mnt/mmc/root/rootfs.tar.bz2 myrootfs/*’
    - run ‘sudo rm –r myrootfs’

- Un-mount the card, run ‘umount /mnt/mmc’

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

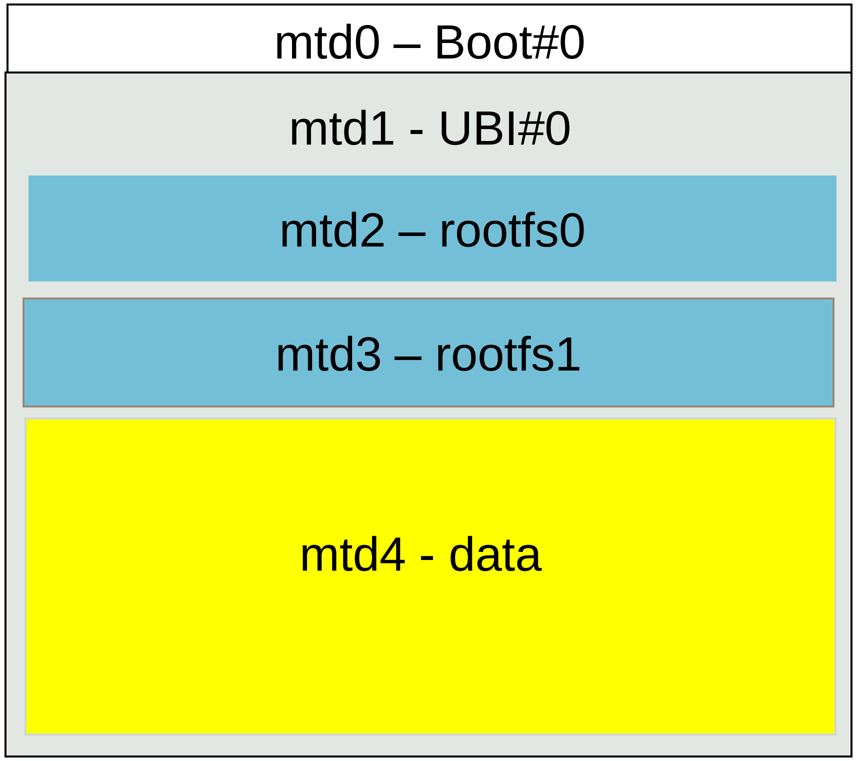
Preparing for Nand boot

The nand driver is MTD system in Linux

- UBI (Unsorted Block Image) system is used to manage the raw flash devices
  - It can manage multiple logical volumes on a single physical devices.
  - It spread the read/write load or manage the wear leveling across the whole chip
  - It consists of consecutive Logical erase blocks, LEBs.
  - Each LEB may be mapped to any physical erase blocks, PEBs.
  - UBI manages the mapping which is hidden from users.
  - UBI also manage bad PEBs. It does not map LEBs to bad PEBs.
  - UBI volume can be create and remove
  - It minimize the chance of loosing data by scrubbing the PEBs
- UBIFS is the file system sitting on top of the UBI managed MTD
- Use the ‘cat /proc/mtd’ to see mtd mapping over the UBI
  - It shows the mtd device name, size, erase size, and the UBI volume name.
- The Nand boot stream is stored in mtd0
- There is a back up rootfs in the iMX233 Bsp so if the rootfs0 is corrupted, the device can use rootfs1.

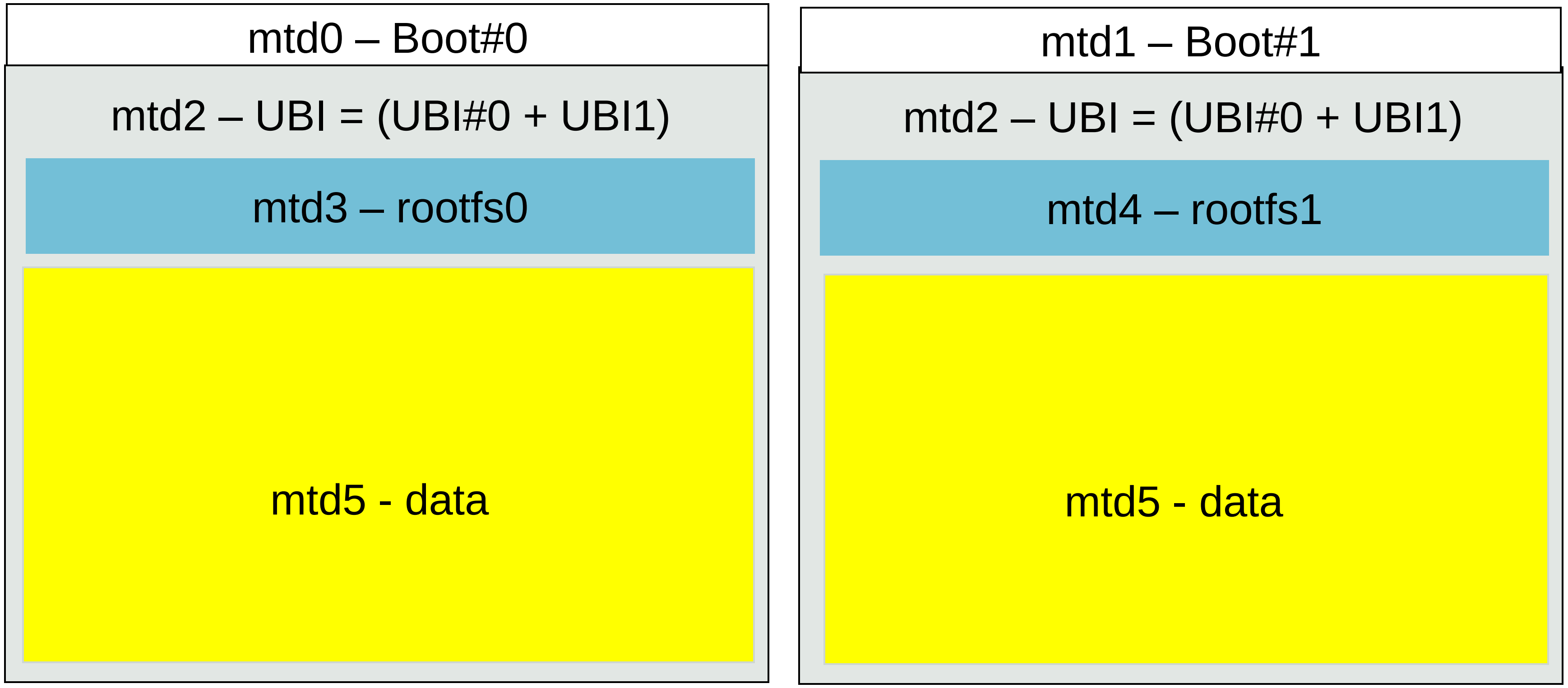
Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Preparing for Nand boot – 1 CS Nand layout



Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Preparing for Nand boot – 2 CS Nand layout



Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Preparing for Nand boot – flash images from SD/MMC

The device is booted to SD/MMC card

- Flash the boot stream to mtd0
  - Run ‘kobs-ng init stmp378x_linux.sb’
    - It stores stmp378x_linux.sb to mtd0 for single CS device
    - It stores stmp378x_linux.sb to mtd0 and mtd1 for dual CS device
      - It combines UBI#0 and UBI#1 to UBI
- Erase the flash (SD/MMC boot
  - Run ‘flash_eraseall /dev/mtd1’ for single CS device
  - Run ‘flash_erasell /dev/mtd2’ for dual CS device
- Attach the ubi manager
  - Run ‘ubiattach /dev/ubi_ctrl –m 1’ for single CS device
  - Run ‘ubiattach /dev/ubi_ctrl –m 2’ for dual CS device
- Create rootfs0 volume, assume the size of rootfs is < 110MiB
  - Run ‘ubimkvol /dev/ubi0 –N rootfs0 –s 110MiB’
- Create rootfs1 volume
  - Run ‘ubimkvol /dev/ubi0 –N rootfs1 –s 110MiB’ for single CS device
  - Run ‘ubimkvol /dev/ubi1 –N rootfs1 –s 110MiB’ for dual CS device
- Create data volume using the rest of the space
  - Run ‘ubimkvol /dev/ubi0 –N data –m’ for single CS device
  - Run ‘ubimkvol /dev/ubi –N data –m’ for dual CS device

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Preparing for Nand boot – flash rootfs from SD/MMC

The device is booted to SD/MMC card (cont)

- Flash the rootfs0
  - Make a ubifs mounting point if there is none
    - Run ‘mkdir /mnt/ubifs’
  - Mount the rootfs0 volume
    - Run ‘mount –t ubifs /ubi0:rootfs0 /mnt/ubifs
  - Unpack the rootfs.tar.bz2 on the SD/MMC card
    - Run ‘tar jxf rootfs.tar.bz2 –C /mnt/ubifs
  - Unmount the ubifs
    - Run ‘umount /mnt/ubifs’
- Flash the rootfs1
  - Mount the rootfs1 volume
    - Run ‘mount –t ubifs /ubi0:rootfs1 /mnt/ubifs’ for single CS device
    - Run ‘mount –t ubifs /ubi1:rootfs1 /mnt/ubifs’ for dual CS device
  - Unpack the rootfs.tar.bz2 to the volume
    - Run ‘tar jxf rootfs.tar.bz2 –C /mnt/ubifs’
  - Unmount the ubifs

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Prepare for Boot over the ethernet

There is similarity between booting over ethernet and USB. The EVK has an ethernet port so we will use NFS over the ethernet

Prepare the TFTP and NFS server. We will use the tftpd and nfs-kernel-server for this example.

- Install tftpd and nfs-kernel-server
  - Run ‘apt-get install tftpd nfs-kernel-server’
  - Make a tftp working directory
    - Run ‘sudo mkdir /srv/tftp’
    - Run ‘sudo ln -s ~/ltib/rootfs/boot/uImage /srv/tftp’ to link the uImage to be used by the tftp server
  - Make a nfs rootfs work area
    - Run ‘sudo mkdir /tools’
    - Run ‘sudo ln –s ~/ltib/roofs /tools’ to link the rootfs from the boot directory
  - Tell the nfs server what to be exports
    - Edit ‘/ect/exports’ file to have ‘/tools/rootfs *(rw,sync,no_root_squash,no_all_squash,no_subtree_check)’ at the end of file.
  - Connect the tartget to the server with a cross-over ethernet cable or two straight cables with a hub in between.
  - Restart the server if necessary
    - Run ‘sudo /etc/inet.d/openbsd.inetd restart’ and/or
    - Run ‘sudo /etc/inet.d/nfs-kernel-server restart’

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Prepare for Boot over the ethernet (cont.)

Flash the u-boot image to the Nand

- Follow the same procedure for flashing the boot stream from SD/MMC card but use the ‘stmp378x_uboot.sb’ instead of ‘stmp378x_linux.sb.
- The u-boot image can also flashed using the UUT which is cover later

Run the uBoot on the target

- Boot the device up in boot mode 4, set boot switch=off,on,off,off
- Power up the EVK.
- Hit ^C to break the auto boot
- Type ‘printenv’ list the u-boot environments set by LTIB
- If necessary change the bootargs to read ‘console=115200n8 ssp1=spi1 ssp2=gpmi enc28j60=0@1,2:3 root=/dev/nfs nfsroot=192.167.10.1:/tools/rootfs,rsize=1024,wsize=1024 ip=192.167.10.2:192.167.10.1::::eth0:off lcd_panel=lms430’

- Type ‘boot’ to resume connection with the host. Restart the server if needed.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Flashing the images using the UUT

Universal Updater tool is an Windows PC application. It communicates with the EVK in the USB HID Class. There are a few problems in the UUT in this release of the LTIB.

It is more convenience to load the UUT from the extranet to a USB drive, update the firmware.sb and packed rootfs file from the Linux Host. The USB drive can easily be used on PC to update the boot stream on the EVK.

Download and unzip the UUT from extranet

- It contains a directory, Universal_updater_Tool, and a sub directory named, files.

Load the boot stream image from LTIB built

- Connect the USB drive to the Linux host
- Copy the Nand boot image to the updater as firmware.sb
  - Run 'sudo cp ~/ltib/rootfs/boot/stmp378x_linux.sb /media/USB/Universal_updater_Tool/firmware.sb'
- If the u-boot is the image to be flash then copy 'stmp378x_uboot.sb' to 'firmware.sb' instead

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Flashing the images using the UUT (cont.)

Load the rootfs image from LTIB built when making the bootstraping to Nand flash

- Copy the rootfs to a temporarily directory and remove the images in the boot directory to reduce the size of the rootfs
  - Change to LTIB working directory, ‘cd ~/ltib’
  - Run ‘sudo cp –a rootfs ,/myrootfs’
  - Run ‘sudo rm ./myrootfs/boot/*’
  - Check to see there is no other directory that is not belong to the rootfs
- Compress the rootfs
  - Run ‘sudo tar jcf /media/USB/Universal_Updater_Tool/files/rootfs.tar.bz2 ./myrootfs/*’
- Remove the tempory rootfs directory
  - Run ‘sudo rm –r .myrootfs’
- Unmount the USB drive and remove it from Linux host

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Flashing the images using the UUT (cont.)

Running the UUT

- Connect the USB drive to a Windows PC
- Double click on ‘UniversalUpdater.exe’ file. This will open the UUT.
- Put the tartget in device recovery mode, boot mode 0
- Connect the EVK to the Windows PC, the UUT will recognize the EVK and connect it with ready message
- Select the single chip select or multichip select depending on the Nand type to be flashed. The EVK uses a single CS nand.
- Click start and wait. It takes a while to flash the nand. The serial debug port should be connected to monitor the progress.
- The flashing of the image may fail if
  - The number of chip select does not match
  - The nand type is not support
  - Bad nand
- There are a couple of xml files in the Universal_Updater_Tool folder that scripts the operation of the UUT. At this point in time, the updater fails to make the correct partition type and caused the Nand boot to fail.
- The boot switches can be changed to boot mode 4 when the serial terminal shows the UTP messages.
- The UUT can also update image for SD/MMC card but it has the same problem making the correct partition type for the second partition on the card.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Boot preps

The boot preps are small codes that run by the ROM before it execute Linux images

- Clock prep makes the iMX clocks suitable for booting
- Power prep prepare the power modes suitable for running linux kernal from either the battery or incoming 5V on the VBUS
- SDRAM prep prepare the EMI for DRAM operations
- These 3 preps are built using the Green Hill tool so there is not source code included in LTIB release
- Linux prep sets up the default kernel command line. It is designed to allow the kernel to have different command line option depending on which button, SW4, SW5, SW6 or no button is pressed when it is running. If there is no button pressed, the first default command line is used. If SW4 button is pressed then the second command line is used, the SW5 or'>>' button chooses the third command line and the SW6 chooses the last command line. The command lines can be edited in file named 'stmp378x_dev.txt' in the linux_prep/cmdlines directory in the boot_stream package. Use the './ltib –m prep –p boot_stream' to unpack the boot_stream package.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

Questions?

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2007.

freescale™
semiconductor